



* **KLA** is a global technology company that develops advanced system for inspecting, measuring and improving the manufacturing of semiconductors.

Their tools help chipmakers detect tiny defects, control production quality and increase efficiency.

KLA combines optics, AI, robotics and software to support every stage of chip production - from raw wafers to final packaging.

It plays a critical role in making sure the chips inside phones, computers, and cars are reliable and high performing.

1. Optical Inspection :-

KLA develops advanced optical system to detect microscopic defects in semiconductor manufacturing. Using laser, high resolution cameras and variable angle lighting, they provide real-time analysis.

Defects are classified automatically, reducing late stage failures and waste.

The system improves itself through continuous learning, enhancing yield and reliability.

This is a critical quality control layer in chip production.

2. AI :-

KLA uses machine learning algorithms to detect complex defects invisible to the human eye.

AI boosts measurement accuracy, optimizes processes and detects anomalies efficiently.

The system learns from millions of data points and make intelligent decisions in real time. This enables predictive maintenance and performance optimization.

3. Robotics :-

KLA designs robotic system that automate wafer handling, positioning and measurement.

These robots use pattern recognition to ensure high precision without human interference.

They integrate seamlessly with other fab systems, enabling continuous production.

The result is fewer errors, faster cycle times and higher throughput.

* Voyager 1035 workflow :-

1. Sample Load & Alignment - The system loads a wafer and performs initial alignment for optimized scanning.

2. A laser beam rapidly scans the wafer surface to detect defects in real time.

3. AI - The DefectWise algorithm identifies and differentiates critical defects from pattern noise.

4. Visual Reporting - Interactive display shows a defect map with location, type, and severity details.

5. Results are integrated into the MES for trend analysis and real time process control.

Jenkins Pipelines and shared libraries. -Topic 2

- Jenkins Pipelines - הרקד דברים ופוסט: build ו deploy

- Declarative Pipelines :-

- גרסה וקוד יחד

- pipeline, agent, stages

- סדר

pipeline {

agent any

stages {

stage ('Build') {

steps {

echo 'Building ...'

}

}

}

}

- Scripted Pipelines :-

- גרסה וקוד יחד, כלול גרסה

- מזהים מזהים

- מזהים מזהים

node {

stage ('Build') {

echo 'Building ...'

}

}

- Jenkinsfile נקרא

המזהים

pipeline

המזהים

agent

המזהים

parameters

Build, Test, Deploy

stages

המזהים

steps

המזהים

environment

המזהים

post

- Shared Library

המזהים

vars/deploy.groovy

המזהים Jenkinsfile

המזהים

Topic 3 : Azure Pipelines

Azure DevOps is Microsoft's CI/CD tool #
yml file contains build, test, deploy etc. configurations #

∴ azure-pipeline.yml #

trigger	: trigger
pr	: pr
parameters	: parameters
variables	: variables
stages	: stages
jobs	: jobs
steps	: steps

(bash, script) runs in job #

Approvals and checks #

Pipelines → Environments → prod → Approvals and checks
Azure DevOps UI ↓

Topic 4 · Introduction to Infrastructure as Code (IaC) and Tools.

מה זה IaC ?

ישנן דרכים רבות להגדיר את המערכת, אך הדרך הנכונה היא להשתמש ב-IaC. זה מאפשר לנו להגדיר את המערכת באמצעות קוד, מה שמאפשר לנו לשלוט במערכת בצורה טובה יותר.

- Terraform :: IaC ב HCL (HashiCorp Configuration Language)
- Pulumi :: IaC ב Python, TypeScript וכו' (משתמשים במעטפת של Node.js)
- Ansible :: דרכי הפעלה (Playbooks) ב YAML (YAML Ain't Markup Language)
- Vault :: אחסון סודות (Secrets) וניהול גיבויים (Backups)
- Consul :: ניהול רשתות (Networking) וזיהוי שירותים (Service Discovery)

Terraform ::

- HCL (HashiCorp Configuration Language) - שפת הגדרת תצורה
- תוכניתור (Planning) - תהליך שבו Terraform בודק את התוכנית והוא מודע לשינויים
- דקלארטיבי (Declarative) - תיאור המצב הרצוי, ולא תהליך ההגעה אליו
- מנגנון אחסון מצב (State) - אחסון המידע על המצב הנוכחי של המערכת
- קובץ state file - קובץ המכיל את המידע על המצב הנוכחי של המערכת

Pulumi ::

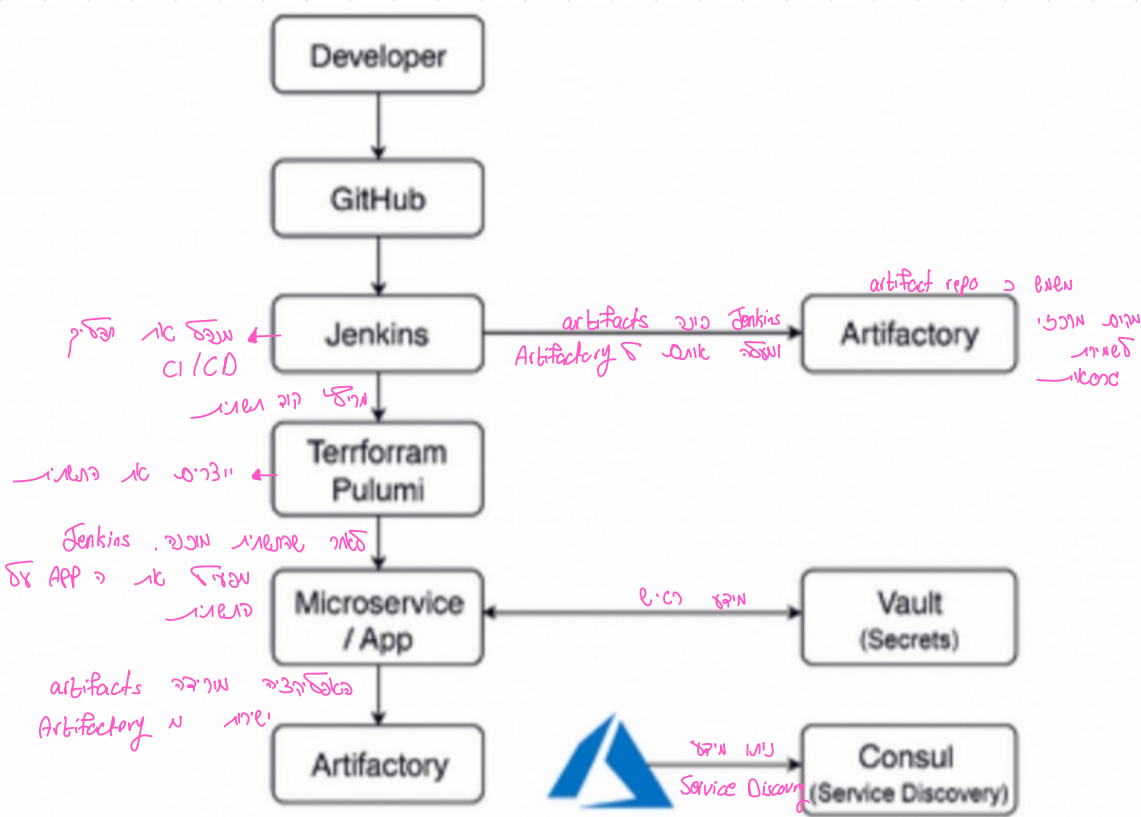
- תוכניתור (Planning) - תהליך שבו Pulumi בודק את התוכנית והוא מודע לשינויים
- דקלארטיבי (Declarative) - תיאור המצב הרצוי, ולא תהליך ההגעה אליו
- מנגנון אחסון מצב (State) - אחסון המידע על המצב הנוכחי של המערכת
- קובץ state file - קובץ המכיל את המידע על המצב הנוכחי של המערכת

Vault ::

- אחסון סודות (Secrets) - אחסון המידע הסודי של המערכת
- ניהול גיבויים (Backups) - ניהול גיבויי המידע הסודי של המערכת
- TLS (Transport Layer Security) - פרוטוקול תקשורת מאובטח

Consul ::

- ניהול רשתות (Networking) - ניהול הרשתות של המערכת
- זיהוי שירותים (Service Discovery) - זיהוי השירותים הנוכחיים של המערכת
- Health Checks - בדיקת הבריאות של השירותים
- Load Balancer - הפצת טעולות בין שירותים



cleanup - policy md

- מיון קבוצות בן יומן נ 30 יום
- מיון קבוצות 5 קבוצות
- מיון קבוצות 5 קבוצות

folder - structure md

- מיון קבוצות בן יומן נ 30 יום
- מיון קבוצות 5 קבוצות
- מיון קבוצות 5 קבוצות

upload.sh - מיון קבוצות בן יומן נ 30 יום

upload.log - מיון קבוצות בן יומן נ 30 יום

Topic 5: Terraform for Windows Projects.

Terraform - יצירת תשתית באמצעות קוד

Modules - תיבות קוד שבהם שומרים קטעי קוד חוזרים

Variables - משתנים שבהם שומרים ערכים שונים

Outputs - תוצאות של תוכנית

Providers - תוכנית שבה נמצאים כל המידע

ADP - תוכנית שבה נמצאים כל המידע
3389 ב' sq כי לאבדו תוכנית

Topic 6 : Putami - Infrastructure as Code with Python

	מאגר המידע	- main - .py
	הקובץ שבו נמצאים כל הקודים	Pulumi.yaml
	הקובץ שבו נמצאים כל הקודים של ה- dev region	Pulumi.dev.yaml
	הקובץ שבו נמצאים כל הקודים של ה- stack	clean up .py

Topic 8: Ansible - Remote Server Automation

laC nimen & w3nGik .80 :- Ansible #

SSH ମିଳନ ଓଲଟି ଯିବ -

—iGnatic- ସମ୍ପର୍କୀୟତା, ପିତୃତ୍ୱ, ମାତୃତ୍ୱ -

- דברים Playbooks שצריך להבין

- המקור של הסיפור הוא באתר של Apache, בעל מידע בקורה את Ansible של דוד כהן.

next Template Engine :- Jinja2

13.8 - 539 55 5555

yaml ile HTML a dosya ekleş -

- Ansible קטן יותר, פשוט יותר, ויש לו יותר תומכים.

[illegible]

2. உறுப்பினர் 2 ஜிங்ஷா. சி. கந்தசாமி

3. `inventory.ini` — פתרון בקבוצה

#	Playbook	הם קובץ	Cong	מיון	שנה	Tasks	e	Ansible	73	כאשר זה	מה נחוץ
---	----------	---------	------	------	-----	-------	---	---------	----	---------	---------

Topic 9 : Packer - Creating Prebuilt Machine Images,

Packer : הוא כלי שיוצר תבניות של VMs (Virtual Machine Images) שניתן להשתמש בהן כדי ליצור VMs על פלטפורמות שונות. הכלי הזה יוצר תבניות של VMs שניתן להשתמש בהן כדי ליצור VMs על פלטפורמות שונות. הכלי הזה יוצר תבניות של VMs שניתן להשתמש בהן כדי ליצור VMs על פלטפורמות שונות.

Builder	מכשיר	הוא	המשתמש	ב	תבנית	של	VM
Provisioners	מכשיר	הוא	המשתמש	ב	תבנית	של	VM

Topic 10: Windows Dockers - Building Containers on Windows.

Windows - מנסים להקים קונטיינרים

Docker - מנסים להקים קונטיינרים

מנסים להקים קונטיינרים

Windows image - Dockerfile

index.html

HTML - Dockerfile

Linux Container

Linux

Linux

(MB) 8

מנסים להקים

Linux Container

Windows Container

Windows OS

Windows

(GB) 20

מנסים להקים

Windows Container

Linux

מנסים להקים

Images

Images

מנסים להקים

Docker Desktop

Topic 11 : Linux Administration and Bash Scripting.

1. monitor_logins.sh

- Write a Bash Script that scans /var/log/syslog
- Filters out all lines with failed login attempts
- Writes a report to /tmp/login_failure.txt

2. User_audit.sh

- Script that lists all users with shell access
- Verifies which ones logged in the past 7 days
- Outputs results in a table format

3. healthcheck.sh

- Check the status of key service
- If a service is inactive, restart it and log the action
- Use systemctl, tee and logger

4. cronjob.txt

- Create a cron job that runs one of the above script every 15
- Log output to a dedict file in /var/log/custom/

minutes

*Topic 12 - Multithreading and Async Programming.

Why we need multithreading? - APIs are not synchronous, they return a response when the request is made, but the response is not ready yet.

∴ Multithreading *

Threads are small units of execution.

Each thread has its own stack and heap.

Async / Await *

Coroutines are a way to write asynchronous code in a synchronous style. They are used to write code that can run asynchronously, but they look like synchronous code.

ThreadPool *

Blocking requests get stuck in a thread.

Threads are expensive to create and destroy.

Asyncio + aiohttp *

Asyncio is a library for asynchronous programming in Python.

It provides a high-level interface for writing asynchronous code.

Topic 13 : Project Development Process + Writing a Test Plan

1. Introduction

Book Management

2. Software Requirements Document

- Introduction

- System Description

- Requirements

- User Requirements

- Functional Requirements

- Boundaries and Constraints

3. Test Plan

- Test Objectives

- Test Scope

- Test Types

- Test Environment

- Test Cases

Boundary conditions, Negative, Positive cases:

Idea → Define Requirements → Write SRS → Design & Plan → Develop Features → Write Test plan → Test Product → Deploy

Smart Books - Create new book, Edit book, Delete book

- Reader

- Admin

- Functional Requirements:

1. Add/Update book entries
2. Track reading status

- Non Functional Requirements

Performance

Security

* Topic 14: Manual Testing - Cloud Infrastructure QA

1. Test Scenario Planning.

test - cases.md

2. Bug Reporting.

DNS resolution, CIDR validation, VPC creation failure

3. End to End Testing

4. API Testing

api - scenarios.json